

Splunk Queries for SOC Analyst



=====

1. Query to identify failed login attempts:

```
sourcetype=auth* "authentication failure"  
| stats count by user  
| sort -count
```

2. Query to identify potential security threats:

```
sourcetype=access_* method=POST status=200 |  
rex field=_raw "password=(?<password>[^&]+)"  
| eval password_length=length(password)  
| where password_length >= 8
```

3. Query to identify privilege escalation attempts:

```
sourcetype=linux_secure su*  
| where user!=root AND user!=""
```

4. Query to identify failed SSH attempts:

```
sourcetype=linux_secure "Failed password for"  
| stats count by src_ip  
| sort -count
```

5. Query to identify successful SSH attempts:

```
sourcetype=linux_secure "Accepted publickey for"  
| stats count by src_ip  
| sort -count
```

6. Query to identify unusual network traffic:

```
sourcetype=network_traffic  
| stats sum(bytes) as total_bytes by src_ip, dest_ip  
| where total_bytes > 1000000
```

7. Query to identify suspicious processes:

```
sourcetype=processes  
| search "lsass.exe" OR "svchost.exe" OR "explorer.exe"  
| stats count by user  
| sort -count
```

8. Query to identify brute force attacks:

```
sourcetype=access_* | stats count by clientip, action | where action="failure" AND count>=5
```

9. Query to identify privilege escalation attempts on Windows systems:

```
sourcetype="WinEventLog:Security" EventCode=4672  
| eval user_account=mvindex(Account_Name,1)  
| search "Security ID" NOT IN ("SYSTEM","LOCAL SERVICE","NETWORK SERVICE")
```

10. Query to identify abnormal user activity:

```
sourcetype=access_* action=purchase  
| stats count by clientip, user  
| where count > 50
```

11. Query to identify potential DNS tunneling activity:

```
sourcetype=dns  
| rex field=answer "data\"\"s*:\s*\"(?<data>[^\"]+)\\"\""  
| eval data_length=len(data)  
| where data_length > 32 AND (data_length % 4) == 0
```

12. Query to identify suspicious PowerShell activity:

```
sourcetype="WinEventLog:Microsoft-Windows-PowerShell/Operational" EventCode=4103  
| eval script_block=mvindex(Message,3)  
| search script_block="*Start-Process*"
```

13. Query to identify unusual file access:

```
sourcetype=access_* action=file_delete OR action=file_rename  
| stats count by user  
| where count > 10
```

14. Query to identify network port scans:

```
sourcetype=network_traffic  
| stats count by src_ip, dest_port  
| where count > 100
```

15. Query to identify suspicious email activity:

```
sourcetype=email  
| search "phishing" OR "malware" OR "suspicious link"
```

16. Query to identify potential data exfiltration:

```
source type=access_* action=file_download  
| stats count by user, dest_ip, dest_port  
| where count > 10
```

17. Query to identify failed VPN attempts:

```
sourcetype=access_* VPN AND action="failure"
```

18. Query to identify successful VPN attempts:

```
sourcetype=access_* VPN AND action="success"
```

19. Query to identify successful login attempts from new or unknown IP addresses:

```
sourcetype=access_* action=login  
| stats count by user, src_ip  
| where count=1
```

20. Query to identify potential SQL injection attempts:

```
sourcetype=access_* method=POST | rex  
field=_raw "SELECT\s+(?<query>[^\s;]+)"  
| eval query_length=length(query)  
| where query_length > 50 AND query_length < 100
```

21. Query to identify unusual file extensions:

```
sourcetype=access_* action=file_upload  
| rex field=file_path ".*\.(?<extension>[^\.]+)"  
| stats count by extension  
| where count > 10
```

22. Query to identify potential phishing attacks:

```
sourcetype=email  
| search "password" OR "reset" OR "verify" OR "login"
```

23. Query to identify traffic to known malicious IP addresses:

```
sourcetype=network_traffic dest_ip=malicious_ip
```

24. Query to identify unusual login times:

```
sourcetype=access_* action=login  
| eval hour=strftime(_time,"%H")  
| stats count by user, hour  
| where count < 3
```

25. Query to identify privilege escalation attempts on Linux systems:

```
sourcetype=linux_secure "sudo:" |  
where user!="root" AND user!=""
```

26. Query to identify potential brute force attacks against a specific user:

```
sourcetype=access_* user=username AND action=failure  
| stats count by src_ip  
| where count >= 5
```

27. Query to identify unusual DNS requests:

```
sourcetype=dns |  
stats count by query  
| where count > 10
```

28. Query to identify potential spear-phishing attempts:

```
sourcetype=email  
| search "CEO" OR "CFO" OR "Finance" OR "Accounting" OR "Payment"
```

29. Query to identify potential malware infections:

```
sourcetype=access_* action=file_download |  
rex field=file_path ".*\.(?<extension>[^\.]*)" |  
search extension="exe" OR extension="dll"
```

30. Query to identify unusual user activity:

```
sourcetype=access_* action=purchase  
| stats count by user  
| where count > 100
```

31. Query to identify potential DDoS attacks:

```
sourcetype=network_traffic  
| stats sum(bytes) as total_bytes by src_ip  
| where total_bytes > 100000000
```

32. Query to identify potential ransomware activity:

```
sourcetype=access_* action=file_delete  
| rex field=file_path ".*\.(?<extension>[^\.]*)"   
| search extension="encrypted" OR extension="locked" OR extension="ransom"
```

33. Query to identify potential insider threats:

```
sourcetype=access_* action=file_upload  
| stats count by user, file_path  
| where count > 10
```

34. Query to identify successful authentication attempts from unknown IP addresses:

```
sourcetype=access_* action=login  
| stats count by src_ip  
| where count >= 5 AND NOT src_ip IN (192.168.0.0/16, 10.0.0.0/8)
```

35. Query to identify potential brute force attacks on a specific service:

```
sourcetype=network_traffic service=ssh  
| stats count by src_ip  
| where count >= 10
```

36. Query to identify successful SSH logins from unusual countries:

```
sourcetype=access_* action=login service=ssh  
| iplocation src_ip  
| stats count by src_country  
| where count > 10 AND NOT src_country="United States"
```

37. Query to identify potential attempts to exploit known vulnerabilities:

```
sourcetype=access_* method=POST  
| rex field=_raw "(?<exploit>CVE-\d{4}-\d+)"  
| stats count by exploit  
| where count > 5
```

38. Query to identify potential brute force attacks on a specific user:

```
sourcetype=access_* user=username AND action=failure  
| stats count by src_ip  
| where count >= 5
```

39. Query to identify potential man-in-the-middle attacks:

```
sourcetype=network_traffic protocol=tcp  
| stats count by dest_ip  
| where count > 100
```

40. Query to identify potential data exfiltration:

```
sourcetype=access_* action=file_upload  
| stats count by user, file_path  
| where count > 10
```

41. Query to identify potential ransomware activity on Windows systems:

```
sourcetype=WinEventLog:Security EventCode=4663 |  
rex field=Object_Name "\\.*\\\\(?!<filename>.+)"  
| rex field=filename ".*\\.(?!<extension>[^\.]+)"  
| search extension="encrypted" OR extension="locked" OR extension="ransom"
```

42. Query to identify unusual network traffic patterns:

```
sourcetype=network_traffic  
| stats count by dest_ip, dest_port  
| where count > 100 AND NOT dest_ip="192.168.0.1"
```

43. Query to identify potential brute force attacks on a specific protocol:

```
sourcetype=network_traffic protocol=http  
| stats count by src_ip  
| where count >= 50
```

44. Query to identify potential account takeover attempts:

```
sourcetype=access_* action=login  
| stats count by user  
| where count > 10
```

45. Query to identify potential DNS tunneling activity:

```
sourcetype=dns  
| stats count by query  
| where count > 5 AND NOT match(query, "\\.")
```

46. Query to identify potential SQL injection attempts on web servers:

```
sourcetype=access_* method=POST uri_path="*.php"  
| rex field=_raw "SELECT\s+(?!<query>[^\;]+)"  
| eval query_length=length(query)  
| where query_length > 50 AND query_length < 100
```

47. Query to identify potential brute force attacks on a specific domain:

```
sourcetype=access_* host=example.com AND action=failure  
| stats count by src_ip  
| where count >= 10
```

48. Query to identify potential brute force attacks on a specific application:

```
sourcetype=access_* uri_path="/app/login" AND action=failure  
| stats count by src_ip  
| where count >= 5
```

49. Query to identify potential phishing attempts through email attachments:

```
sourcetype=email  
| search attachment="*.exe" OR attachment="*.zip"
```

50. Query to identify potential exploitation attempts on vulnerable services:

```
sourcetype=network_traffic  
| stats count by src_ip, dest_port  
| where count > 10 AND dest_port IN (22, 3389, 1433, 3306, 8080)
```

51. Query to identify potential reconnaissance activity:

```
sourcetype=access_* method=GET  
| stats count by uri_path  
| where count > 100
```

52. Query to identify potential cross-site scripting (XSS) attacks on web servers:

```
sourcetype=access_* method=POST uri_path="*.php"  
| rex field=_raw "document\.write\('(?(?<payload>[^\']+)'\)"  
| search payload="<script>"
```

53. Query to identify potential privilege escalation attempts:

```
sourcetype=access_* action=privilege_escalation  
| stats count by user  
| where count > 5
```

54. Query to identify potential web application attacks:

```
sourcetype=access_* method=POST uri_path="*.php"  
| rex field=_raw "(?<attack>sql_injection|xss|csrf)"  
| stats count by attack  
| where count > 5
```

55. Query to identify potential lateral movement attempts:

```
sourcetype=network_traffic protocol=tcp dest_port=445  
| stats count by src_ip, dest_ip  
| where count > 10
```

56. Query to identify potential unauthorized changes to critical files:

```
sourcetype=access_* action=file_write  
| search file_path="*/etc/*" OR file_path="*/var/*"
```

57. Query to identify potential port scanning activity:

```
sourcetype=network_traffic protocol=tcp  
| stats count by src_ip, dest_port  
| where count > 20 AND NOT dest_port IN (22, 3389, 1433, 3306, 8080)
```

58. Query to identify potential malicious PowerShell activity on Windows systems:

```
sourcetype=WinEventLog:Windows PowerShell EventCode=4104  
| search (New-Object System.Net.WebClient).DownloadString OR (Invoke-WebRequest -Uri)
```

59. Query to identify potential SQL injection attempts on web servers:

```
sourcetype=access_* method=POST uri_path="*.php"  
| rex field=_raw "SELECT\s+(?<query>[^\s;]+)"  
| eval query_length=length(query)  
| where query_length > 100 AND query_length < 200
```

60. Query to identify potential brute force attacks on a specific domain controller:

```
sourcetype=WinEventLog:Security EventCode=4625 domain_controller="DC01"  
| stats count by src_ip  
| where count >= 5
```

61. Query to identify potential DDoS attacks:

```
sourcetype=network_traffic  
| stats count by src_ip  
| where count > 1000
```

62. Query to identify potential web shell activity:

```
sourcetype=access_* action=command_execution  
| search (echo|print|printf)\s+(base64_decode|eval|gzinflate|str_rot13)
```

63. Query to identify potential brute force attacks on a specific network device:

```
sourcetype=cisco:asa |  
stats count by src_ip  
| where count >= 10
```

64. Query to identify potential privilege escalation attempts on Linux systems:

```
sourcetype=access_* action="sudo command"  
| stats count by user  
| where count >= 10
```

65. Query to identify potential DNS tunneling activity:

```
sourcetype=dns  
| rex field=_raw "\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}#(?<query>.+)\s+(\d+)\s+type:  
(?<type>.+)\s+class: (?<class>.+)\s+[\d\s]+flags: (?<flags>.+)\s+;[\s\S]+response:\s+no error"  
| search type="A" AND class="IN" AND flags="rd"
```

66. Query to identify potential lateral movement attempts using RDP:

```
sourcetype=WinEventLog:Security EventCode=4624 OR EventCode=4625  
| search Logon_Type=10
```

67. Query to identify potential command and control (C2) traffic:

```
sourcetype=network_traffic  
| stats count by dest_ip  
| where count > 500 AND NOT dest_ip IN (192.168.0.0/16, 10.0.0.0/8)
```

68. Query to identify potential PowerShell Empire activity:

```
sourcetype=WinEventLog:Windows PowerShell  
| search (powershell.exe -nop -w hidden -ep bypass -c)|(iex(new-object  
net.webclient).downloadstring)
```

69. Query to identify potential ransomware activity:

```
sourcetype=access_* action=file_write  
| search file_path="*.crypt" OR file_path="*.locky"
```

70. Query to identify potential malicious traffic from a specific IP address:

```
sourcetype=network_traffic src_ip=10.1.1.1  
| stats count by dest_ip  
| where count > 10
```

71. Query to identify potential brute force attacks on web applications:

```
sourcetype=access_* method=POST uri_path="*.php"  
| stats count by src_ip  
| where count >= 50
```

72. Query to identify potential unauthorized access attempts to sensitive files:

```
sourcetype=access_* action=file_read  
| search file_path="*/etc/shadow" OR file_path="*/etc/passwd"
```

73. Query to identify potential lateral movement attempts using SMB:

```
sourcetype=WinEventLog:Security EventCode=5140  
| search Object_Name="*\\ADMIN$" OR Object_Name="*\\C$"
```

74. Query to identify potential brute force attacks on SSH servers:

```
sourcetype=linux_secure action=invalid  
| stats count by src_ip  
| where count >= 10
```

75. Query to identify potential phishing attacks:

```
sourcetype=access_* method=POST uri_path="*.php"  
| search form_action="http://www.evilsite.com/login.php" AND (input_password=* OR  
input_password=*)
```

76. Query to identify potential command injection attempts on web servers:

```
sourcetype=access_* method=POST uri_path="*.php"  
| rex field=_raw "(?<command>cat|ls|dir)\s+(?<argument>[^\s;]+)"  
| where isnotnull(command) AND isnotnull(argument)
```

77. Query to identify potential lateral movement attempts using WinRM:

```
sourcetype=WinEventLog:Microsoft-Windows-WinRM/Operational EventCode=146  
| search "winrs: client" AND "is starting a command" AND NOT user="NETWORK SERVICE" AND  
NOT user="LocalSystem"
```

78. Query to identify potential brute force attacks on FTP servers:

```
sourcetype=access_* method=POST uri_path="*/wp-login.php"  
| stats count by src_ip  
| where count >= 20
```

79. Query to identify potential privilege escalation attempts on Windows systems:

sourcetype=WinEventLog:Security EventCode=4688

| search (New_Process_Name="*\\runas.exe" OR New_Process_Name="*\\psexec.exe") AND NOT User="SYSTEM"

80. Query to identify potential beaconing activity from a compromised host:

sourcetype=network_traffic src_ip=10.1.1.1

| stats count by dest_port

| where count > 1000

81. Query to identify potential brute force attacks on SSH servers (failed login attempts):

sourcetype=linux_secure action=failed

| stats count by src_ip

| where count >= 10

82. Query to identify potential data exfiltration attempts over HTTP:

sourcetype=access_* action=file_download

| search uri_path="*.zip" OR uri_path="*.rar" OR uri_path="*.tgz" OR uri_path="*.tar.gz"

83. Query to identify potential lateral movement attempts using WMI:

sourcetype=WinEventLog:Security EventCode=5861

| search (Operation="ExecQuery" AND QueryLanguage="WQL") OR (Operation="MethodCall" AND NOT MethodName="GetSecurityDescriptor" AND NOT MethodName="SetSecurityDescriptor")

84. Query to identify potential brute force attacks on MSSQL servers:

sourcetype=mssql_access action=failed

| stats count by src_ip

| where count >= 10

85. Query to identify potential privilege escalation attempts using PowerShell:

sourcetype=WinEventLog:Microsoft-Windows-PowerShell/Operational EventCode=400

| search "PowerShell pipeline execution details" AND NOT "UserPrincipalName=SYSTEM@*" AND NOT "UserPrincipalName=NETWORK SERVICE@*"

86. Query to identify potential brute force attacks on email accounts:

sourcetype=exchangepops

| stats count by src_ip

| where count >= 10

87. Query to identify potential lateral movement attempts using RDP (successful logins):

sourcetype=WinEventLog:Security EventCode=4624

| search Logon_Type=10

88. Query to identify potential brute force attacks on MSSQL servers (successful logins):

sourcetype=mssql_access action=success

| stats count by src_ip

| where count >= 10

89. Query to identify potential data exfiltration attempts over FTP:

sourcetype=access_* action=file_upload

| search uri_path="*/ftp" OR uri_path="*/sftp"

90. Query to identify potential lateral movement attempts using SMB (successful connections):

sourcetype=WinEventLog:Security EventCode=5140

| search Object_Name="*\\ADMIN\$" OR Object_Name="*\\C\$"

91. Query to identify potential brute force attacks on RDP:

sourcetype=WinEventLog:Security EventCode=4625

| search Logon_Type=10 AND Status="0xC000006D"

92. Query to identify potential brute force attacks on web applications:

sourcetype=access_* method=POST

| stats count by src_ip, uri_path

| where count >= 100

93. Query to identify potential lateral movement attempts using Remote Registry Service:

sourcetype=WinEventLog:Security EventCode=4663

| search Object_Name="*\\REGISTRY\\MACHINE\\SOFTWARE" AND NOT User="SYSTEM" AND NOT User="NETWORK SERVICE" AND NOT User="LOCAL SERVICE"

94. Query to identify potential privilege escalation attempts on Linux systems (sudo usage):

sourcetype=linux_secure "sudo:"

95. Query to identify potential data exfiltration attempts over DNS:

sourcetype=dns

| search query_type=A AND query != "*.google.com" AND query != "*.facebook.com" AND query != "*.twitter.com" AND query != "*.microsoft.com"

96. Query to identify potential lateral movement attempts using SMB (failed connections):

sourcetype=WinEventLog:Security EventCode=5152

| search Object_Name="*\ADMIN\$" OR Object_Name="*\C\$" AND Status="0xC000006D"

97. Query to identify potential brute force attacks on MSSQL servers (failed logins):

sourcetype=mssql_access action=failed

| stats count by src_ip

| where count >= 10

98. Query to identify potential data exfiltration attempts over SMTP:

sourcetype=smtp action=send_message

| search recipient!="*@gmail.com" AND recipient!="*@yahoo.com" AND

recipient!="*@hotmail.com" AND recipient!="*@aol.com"

99. Query to identify potential lateral movement attempts using NetBIOS:

sourcetype=WinEventLog:Security EventCode=5719

| search "No Domain Controller is available" OR "This computer was not able to set up a secure session with a domain controller"

100. Query to identify potential brute force attacks on Telnet servers:

sourcetype=access_* method=POST uri_path="*/telnet"

| stats count by src_ip

| where count >= 10

101. Query to identify potential data exfiltration attempts over FTP:

sourcetype=ftp action=putfile

| stats count by src_ip

| where count >= 10

102. Query to identify potential lateral movement attempts using WMI (failed connections):

sourcetype=WinEventLog:Security EventCode=5605

| search Object_Name="*\ROOT\cimv2" AND NOT User="SYSTEM"

103. Query to identify potential brute force attacks on SSH servers:

sourcetype=access_* method=POST uri_path="*/ssh"

| stats count by src_ip

| where count >= 10

104. Query to identify potential privilege escalation attempts on Windows systems (services configuration changes):

```
sourcetype=WinEventLog:Security EventCode=4697 OR EventCode=7045  
| search Image_Path="*\\System32\\" AND NOT User="SYSTEM"
```

105. Query to identify potential brute force attacks on SNMP:

```
sourcetype=snmptrap |  
stats count by src_ip  
| where count >= 10
```

106. Query to identify potential data exfiltration attempts over HTTP:

```
sourcetype=access_* method=POST uri_path="/upload"  
| stats count by src_ip  
| where count >= 10
```

107. Query to identify potential lateral movement attempts using DCOM (failed connections):

```
sourcetype=WinEventLog:Security EventCode=10009  
| search "DCOM was unable to communicate with the computer" AND NOT User="SYSTEM"
```

108. Query to identify potential brute force attacks on MySQL servers:

```
sourcetype=mysql_access action=failed  
| stats count by src_ip  
| where count >= 10
```

109. Query to identify potential privilege escalation attempts on Windows systems (scheduled tasks creation):

```
sourcetype=WinEventLog:Security EventCode=4698  
| search "Task Scheduler service found a misconfiguration" AND NOT User="SYSTEM"
```

110. Query to identify potential data exfiltration attempts over HTTPS:

```
sourcetype=ssl method=POST  
| stats count by src_ip, dest_ip  
| where count >= 10
```